

# Motor Fault Diagnosis Based on Short-time Fourier Transform and Convolutional Neural Network

Li-Hua Wang<sup>1</sup> · Xiao-Ping Zhao<sup>2,3</sup> · Jia-Xin Wu<sup>2</sup> · Yang-Yang Xie<sup>1</sup> · Yong-Hong Zhang<sup>1</sup>

Received: 19 January 2017/Revised: 22 August 2017/Accepted: 29 September 2017/Published online: 15 November 2017  
© Chinese Mechanical Engineering Society and Springer-Verlag GmbH Germany 2017

**Abstract** With the rapid development of mechanical equipment, the mechanical health monitoring field has entered the era of big data. However, the method of manual feature extraction has the disadvantages of low efficiency and poor accuracy, when handling big data. In this study, the research object was the asynchronous motor in the drivetrain diagnostics simulator system. The vibration signals of different fault motors were collected. The raw signal was pretreated using short time Fourier transform (STFT) to obtain the corresponding time-frequency map. Then, the feature of the time-frequency map was adaptively extracted by using a convolutional neural network (CNN). The effects of the pretreatment method, and the hyper parameters of network diagnostic accuracy, were investigated experimentally. The experimental results showed that the influence of the preprocessing method is small, and that the batch-size is the main factor affecting accuracy and training efficiency. By investigating feature visualization, it was shown that, in the case of big data, the extracted CNN features can represent complex mapping relationships between signal and health status, and can also overcome the prior knowledge and engineering experience

requirement for feature extraction, which is used by traditional diagnosis methods. This paper proposes a new method, based on STFT and CNN, which can complete motor fault diagnosis tasks more intelligently and accurately.

**Keywords** Big data · Deep learning · Short-time Fourier transform · Convolutional neural network · Motor

## 1 Introduction

Motors have been widely used as key machine components for the production of torque. Any motor failure will cause unwanted downtime, expensive repair procedures, and even human casualties. As an effective component of condition-based maintenance, fault diagnosis has gained much attention to guarantee safe motor operations [1].

Motor conditions can be reflected by vibratory [2], acoustic [3], thermal [4], and electrical [5] measurements, among others. To fully inspect the health condition of motors, condition monitoring systems are used to collect real-time data from machines, therefore, high amounts of data are acquired after prolonged motor operation [6]. As the data is generally collected faster than diagnosticians can analyze it, there is an urgent need for diagnosis methods that can effectively analyze massive amounts of data and provide accurate diagnosis results automatically. These types of methods are called intelligent fault diagnosis methods. Glowacz [7] proposed a motor fault analysis technique for acoustic signals using the Coiflet wavelet transform and K-nearest neighbor classifier. Zhao, et al. [8] used the wavelet analysis method for decomposing the vibration

Supported by National Natural Science Foundation of China (Grant No.51405241, 51505234, 51575283).

✉ Xiao-Ping Zhao  
zxp@nuist.edu.cn

<sup>1</sup> School of Information and Control, Nanjing University of Information Science and Technology, Nanjing 210044, China

<sup>2</sup> School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China

<sup>3</sup> Network Monitoring Center of Jiangsu Province, Nanjing University of Information Science and Technology, Nanjing 210044, China

acceleration signal of the motor, to obtain the energy ratio of each sub-frequency band. Then, they used the energy ratio to train the optimized support vector machine (SVM). Li, et al. [9] proposed a fault diagnosis method for an asynchronous motor, which was based on kernel principal component analysis and particle swarm SVM. For other diagnostic objects, Pandya, et al. [10] utilized multinomial logistic regression and wavelet packet transform to diagnose bearing faults. Khazaei, et al. [11] developed a fault classifier using the fusion of vibration data and acoustic signals for planetary gearboxes based on the Dempster-Shafer evidence theory. However, some obvious deficiencies were discovered by carrying out a literature review. The features input to the classifiers were extracted and selected by diagnosticians from measured signals largely depending on prior knowledge about signal processing techniques and diagnostic expertise. In addition, manual feature extraction often makes raw signals lose a certain part. Thus, it is necessary to adaptively mine the characteristics hidden in measured signals to reflect the different health conditions of the machinery, instead of manually extracting and selecting features.

Deep learning has the potential to overcome the aforementioned deficiencies in current intelligent diagnosis methods. In 2006, Hinton, et al. [12] proposed a deep learning method for the first time, and it set off a wave of interest in deep learning, in the academic and industrial fields. Presently, deep learning shows a clear advantage in processing large data volumes of images and speech [13]. Krizhevsky, et al. [14] developed a DNN-based method, in a large-scale visual recognition challenge, which involved millions of labeled images, and obtained the best results. In 2012, Hinton, et al. [13] made significant progress in speech recognition using deep neural networks, and the training data reached 3000 h. The aforementioned applications prove that deep learning is a promising tool in dealing with massive amounts of data. Deep learning has also been applied in the field of mechanical fault diagnosis. Li, et al. [15] utilized singular value decomposition and deep belief networks in building a fault diagnosis system for rolling bearings. The system achieved a satisfactory result. Feng, et al. [16] proposed a new method for gear fault diagnosis. Using this method, they established a stacked auto-encoder network and then utilized the frequency domain as input, to train the network and realize gear fault diagnosis. Considering the similarity between the health states of complex rotary machinery components and heterogeneous data, in image pattern classification problems with high-dimensionality, deep learning methods may show considerable potential in system fault diagnosis, with respect to the advantage of a dominant training

mechanism and deep learning architecture [17]. In addition, deep learning is thought capable of discovering useful high-order feature representations, as well as the relevance of raw signals, which motivate the emergence of promising applications for addressing, effectively and accurately, diagnosis problems encountered during classification tasks with complex and mixed system health states [18]. Recent theoretical studies have also explored the concept of deep hierarchical architectures needing to yield a new point for complex distributions, to achieve a better and more robust generalization performance in challenging recognition tasks [19, 20]. However, although there exists considerable potential, as well as a crucial need to address these challenges by utilizing the advantages of deep learning techniques, these are still rarely applied in current fault diagnosis research of electromechanical systems [21].

In this study, a deep learning method based on short-time Fourier transform (STFT) [22] and a convolutional neural network (CNN) is proposed with respect to complex sensory signals and ambient influence. The raw signal was converted into a time frequency map using STFT. Subsequently, the time frequency map was used as input to the CNN, where it utilized these preprocessed samples for carrying out supervised training to realize motor fault diagnosis. The proposed deep learning methods were validated using testing datasets. The fault diagnosis accuracy of the proposed deep learning method can be used to form a knowledge base for determining if the approach is applicable to detecting and classifying the health states of complex systems with inevitable interference. Existing health state classification methods, such as SVM, were used for comparison.

This paper is organized as follows: CNN methods are introduced in Section 2. In Section 3, a description of data preprocessing and model design is provided. In Section 4, the proposed model is validated using test datasets collected from the drivetrain diagnostics simulator system. Moreover, in this section, an investigation of parameter selection and feature visualization is discussed, and a comparison to other methods is made. In Section 5, the paper is concluded.

## 2 Convolutional Neural Network

A CNN is a recently developed and highly effective recognition method, which has attracted much attention. A CNN can input the original image directly and avoid its complicated pretreatment. Additionally, a CNN is highly invariant to image information in the form of translation, scaling, inclination, or other deformation, owing to its local receptive field, weight sharing, and down sampling.

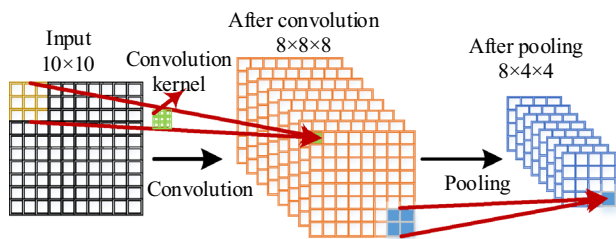


Figure 1 Sub-convolutional neural network

Therefore, it has been widely used because of the above mentioned advantages.

A CNN comprises multiple sub-convolutional neural networks (sub-CNNs, as shown in Figure 1). The network consists of a set of layers, each of which contains one or more planes. In the first sub-CNN, pre-processed images are entered in the input layer. To other sub-CNNs, the output feature map of the previous layer will be the input of the next sub-CNN. The output feature map of the last sub-CNN is connected to the fully connected layer and the classifier, which is used for the recognition of images, speech, and so on.

### 2.1 Convolutional Layer

Natural images have inherent features, such as features obtained from an image sub-block, after learning. These features can be applied as a filter to all sub-blocks. Then, we can obtain the activation values of different sub-blocks. The convolution in a CNN uses these inherent image features.

There are two important basic concepts in convolutional computation; namely, local receptive fields and shared weights. In the CNN, we take the input layer as a two-dimensional matrix, such as the input in Figure 1. Each unit in a plane receives input from a small neighborhood in the planes of the previous layer. The weights forming the receptive field for a plane are forced to be equal at all points in the plane. Each plane can be considered a feature map with a fixed feature detector convolved with a local window, which is scanned over the planes in the previous layer. Multiple planes are usually used in each layer so that multiple features can be detected. These layers are called convolutional layers [23].

In Figure 1, eight 3 × 3 convolutional kernels are used to convolute the 10 × 10 input feature map. Eight feature maps, with a size of 8 × 8, were obtained. The general form of the convolution operation is as expressed by Eq. (1).

$$x = f\left(\sum x * w_{ij} + b\right), \tag{1}$$

where, \* stands for the operator of the two-dimensional discrete convolution, **b** is the bias vector,  $w_{ij}$  and **x** denote

the convolution kernel and the input feature map, respectively.  $f\{\cdot\}$  represents the activation function.

### 2.2 Pooling

After the processing of the convolutional layer, the number of feature images is increased, making the feature dimension very large. This makes it easy to cause the curse of dimensionality. Therefore, to solve this problem, we used aggregate statistics for the feature maps obtained by the convolutional layer. Accordingly, it became more convenient to describe the high dimensional image. This aggregation operation is called pooling. The pooling operation reduces the resolution of the output feature map, and can still better retain the features extracted from the high-resolution feature maps. The general form of the down sampling is expressed by Eq. (2).

$$x = f(\beta \text{down}(x) + b), \tag{2}$$

where,  $\beta$  is the multiplicative bias term,  $\text{down}(x)$  is the pooling function, **b** is the additive bias vector, and  $f(\cdot)$  is the activation function.

As shown in Figure 1, the eight 8 × 8 feature maps are obtained by the convolution of input feature maps. The eight 4 × 4 feature maps are obtained after pooling so that the dimension of the feature maps is reduced.

### 2.3 Fully Connected Layer

All the neurons in the fully connected layer are connected to all neurons in the feature maps of the upper layer, whose output is expressed by Eq. (3).

$$h(x) = f(wx + b), \tag{3}$$

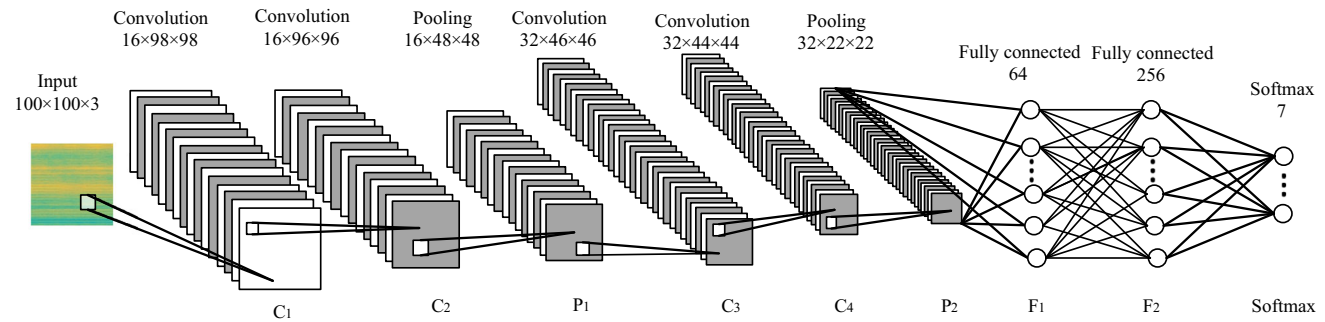
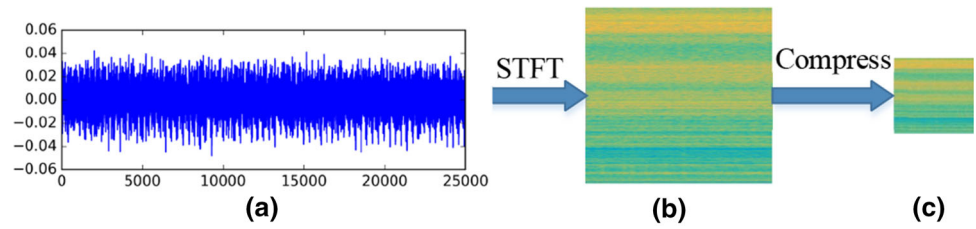
where, **x** is the input of the fully connected layer,  $h(x)$  represents the output of the fully connected layer, **w** and **b** denote the weight and the additive bias term, and  $f(\cdot)$  represents the activation function.

To prevent over fitting in the classification, the “dropout” method, in the fully connected layer, is usually introduced. In the training, we normally let some neurons in the hidden layer stop working, with a certain probability *P*, to improve the generalization ability and prevent over fitting.

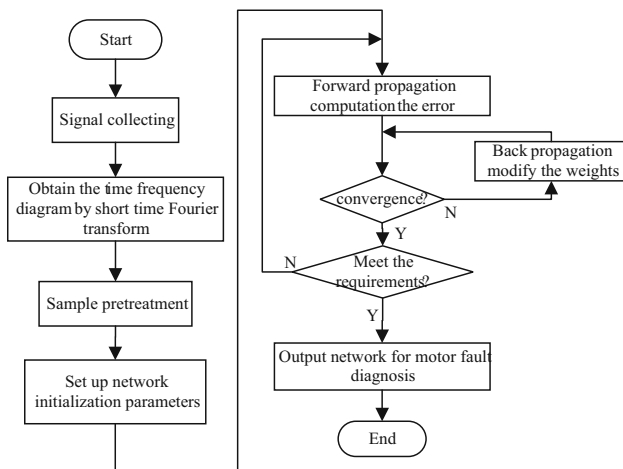
### 2.4 Classifier

Softmax [24] is the generalization of the logistic classifier, mainly for solving the multi-classification problem. If we suppose that the input sample in the training data is *x* and that the corresponding label is *y*, it will determine the sample *j* probability for the category  $p(y = j|x)$ . Thus, for a class *K* classifier, the output will be a vector of the *K*-dimension (the sum of elements in a vector is 1), as shown by Eq. (4).

**Figure 2** Pretreatment of original signal



**Figure 3** Convolutional neural network



**Figure 4** Flow chart of motor fault diagnosis

$$h_{\theta}(x^{(i)}) = \begin{bmatrix} p(y^{(i)} = 1|x^{(i)}; \theta) \\ p(y^{(i)} = 2|x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k|x^{(i)}; \theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \vdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix}, \quad (4)$$

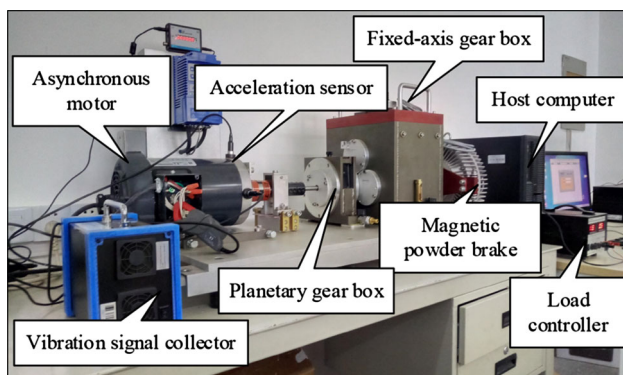
where,  $\theta_1; \theta_2; \dots; \theta_k; \in \mathfrak{R}^{n+1}$  are the parameters of our model. Note that the term  $\frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}}$  normalizes the distribution, so that it sums to one.

In the training, after using the gradient descent method, the cost function of Softmax can be minimized by several iterations, to complete network training. The cost function  $J(\theta)$  is expressed by Eq. (5).

$$J(\theta) = -\frac{1}{m} \left[ \sum_{i=1}^m \sum_{j=1}^k 1\{y^i = j\} \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}} \right], \quad (5)$$

where  $1\{\cdot\}$  is an indicative function, which means that, when the value of the braces is true, the result is 1; otherwise, the result is 0.

When training the CNN, the most common approach is to use the back-propagation rule and supervised training methods. The error term is generated based on comparison between the given label and the classifier's output. According to the concept of back-propagation, the error can be transferred to each node, layer by layer, and we can also update the weight (the specific weight updating equation is thoroughly explained in Ref. [25]). The error



**Figure 5** Drivetrain diagnostics simulator system

term becomes smaller and smaller, through repeated iteration, and the weights also become increasingly more stable, for the network to be trained.

### 3 Data Preprocessing and Model Design

Under this method, we have to pre-process the raw signal and convert it to an image format. The input of the CNN should be a matrix of type  $[m, n, k]$ . In the field of image processing, the  $k$  value is usually 3 and represents the three channels of the color image. To deal with the one-dimensional signal using the CNN method, it is necessary to carry out data preprocessing and convert the signal into an  $[m, n, k]$  matrix. Therefore, the conversion of a signal into a time-frequency map is an alternative approach. The general time-frequency analysis method includes STFT, wavelet transformation, and so on. In this paper, STFT is used for time-frequency analysis. In the comparative analysis section, another pretreatment method called wavelet transform [26] is compared to STFT. The specific operation is as shown below.

First, the STFT is used to convert the signal, shown in Figure 2(a), into a time-frequency map, as shown in Figure 2(b). Subsequently, we can obtain the rectangle, shown in Figure 2(b). To reduce the amount of calculation and facilitate the training of the CNN, the map is compressed into a  $100 \times 100$  square, as shown in Figure 2(c).

After simple pretreatment, the time-frequency map can be used to train the neural network. In the CNN model, as shown in Figure 3,  $C_1, C_2, C_3,$  and  $C_4$  are convolutional layers (the convolution kernel size used in the convolutional layer is  $3 \times 3$ );  $P_1$  and  $P_2$  are down sampling layers (the down sampling layer uses the maximum down sampling method, with the sampling unit being  $2 \times 2$ ).  $F_1$  and  $F_2$  are the fully connected layers and Softmax is the classification layer.

The motor fault diagnosis process is shown in Figure 4.

## 4 Experiment and Analysis

### 4.1 Data Description

The motor data used in these experiments were collected from the asynchronous motor in the drivetrain diagnostics simulator system (Figure 5). It contains a two-stage planetary gearbox, two-stage fixed-axis gear box, 3-hp motor for driving the gearboxes, and magnetic brake for loading. The acceleration sensor was installed on the motor and used to acquire vibration signals at a sampling frequency of 5.12 kHz.

**Table 1** Seven motor states

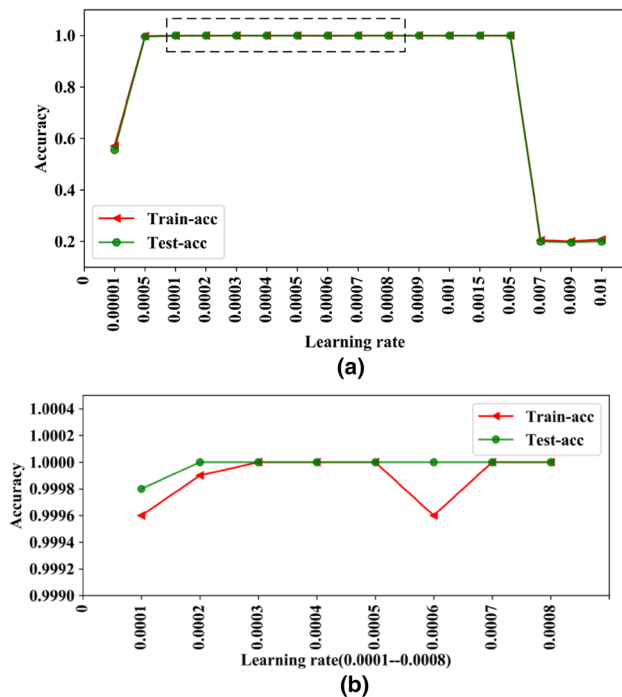
Motor state	CNN sample number	Label	One hot coding
Normal	2000	1	1000000
Built-in rotor imbalance	2000	2	0100000
Stator winding faults	2000	3	0010000
Built-in faulted bearing	2000	4	0001000
Built-in bowed rotor	2000	5	0000100
Built-in broken rotor bars	2000	6	0000010
Voltage imbalance and single phase	2000	7	0000001

**Table 2** Loss and accuracy under different learning rates

Learning rate	Train-loss	Test-loss	Train-acc	Test-acc
0.00001	1.3525	1.3109	0.5709	0.5562
0.00005	0.0625	0.0442	0.98	0.9962
0.0001	0.0137	0.0107	0.9996	0.9998
0.0002	0.0031	0.0033	0.9999	1
0.0003	0.0018	0.0016	1	1
0.0004	0.0012	0.0012	1	1
0.0005	0.00091	0.00056	1	1
0.0006	0.0015	0.00045	0.9995	1
0.0007	0.00065	0.0004	1	1
0.0008	0.00038	0.00035	1	1
0.0009	0.0003	0.00022	1	1
0.001	0.00026	0.00028	1	1
0.0015	0.00018	0.00018	1	1
0.005	0.000048	0.00018	1	1
0.007	1.6101	1.6097	0.2042	0.1994
0.009	1.6109	1.611	0.2002	0.1962
0.01	1.6101	1.6114	0.2072	0.2

Seven experiments were carried out under different motor health conditions (given in Table 1). These conditions involve normal, built-in rotor imbalance, stator winding faults, built-in faulted bearing, built-in bowed rotor, built-in broken rotor bars, and voltage imbalance and single phasing. In each experiment, a damaged motor was installed inside the test rig and the other components were normal.

Rotor imbalance was achieved by taking a balanced rotor from the manufacturer and intentionally removing the balance weights and/or by adding weight. The balance weights were attached to small aluminum pins protruding from both rotor ends. The bowed rotor motor consisted of a motor with an intentionally bent rotor in the center. The faulted bearing motor consisted of a motor with intentionally faulted bearings: one bearing with an inner race fault and one bearing with an outer race fault. The broken



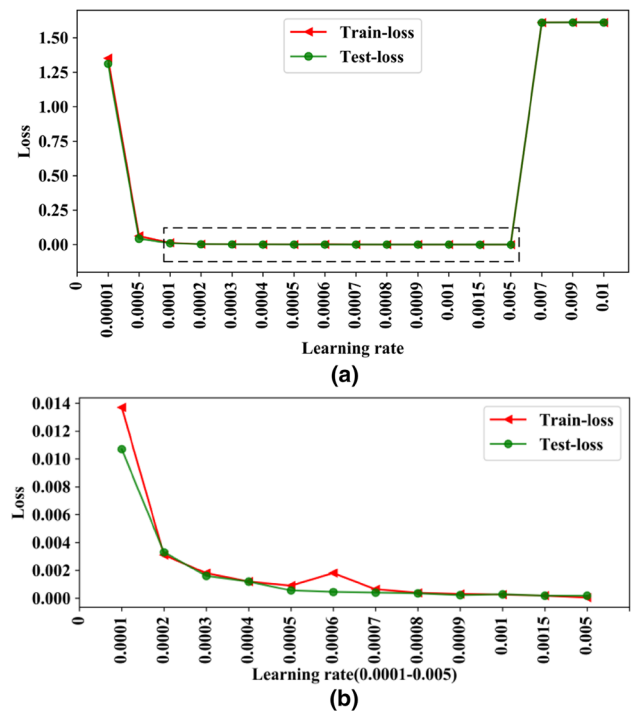
**Figure 6** Loss under different learning rates; **b** is the enlargement of the dotted line in (a)

rotor bar motor consisted of a motor already fitted with an intentionally broken rotor bar. Enough material was removed to expose three rotor bars. The motor had two windings, with a voltage difference ranging from 4 V to 5 V, which was tapped to allow the addition of an extra load to the winding, via an external control box. The control box consisted of a 0–4  $\Omega$  variable resistor. The variable resistor, or rheostat, was used to introduce varying amounts of resistance, in the turn-to-turn short, between the windings. High resistance simulated an insulated winding, while low resistance simulated a shorted winding. Phase loss and voltage imbalance was achieved by switching phases on and off, and by introducing resistance using a control box, by simply connecting the wire from the motor controller to the control box, and then by connecting the control box to the machinery fault simulator. The phase loss switch opened the circuit to the first phase; the voltage control switch introduced a variable resistor of 0–25  $\Omega$  to the second phase; the third phase wiring remained untouched.

Samples amounting to 2000 were collected for each fault in the CNN method, with the length of each signal being 5 s. We randomly selected 20% of them as test samples, and the rest as a training samples.

## 4.2 Selection of CNN Parameters

When training the CNN, it is important to choose the right parameters. The parameters are different for different



**Figure 7** Accuracy under different learning rates; **b** is the enlargement of the dotted line in (a)

sample sets. Adjusting the parameters to find the appropriate parameters of the corresponding sample set is an important link in the CNN training process.

### 4.2.1 Learning rate

In the process of training the CNN, the gradient descent method was used for optimization. The learning rate is an important parameter, which influences the adjustment of weights and error convergence. To improve the efficiency of network training, it is very important to select a suitable learning rate. In this experiment, different learning rates were used to train the CNN, and we obtained different loss and accuracy as shown in Table 2.

From Figures 6 and 7, we can see that, if the learning rate is too large or too small, training and testing accuracy will be reduced with the CNN method. On the contrary, by choosing an appropriate learning rate, we can speed up the convergence of the CNN network and improve its accuracy. In this experiment, the optimized learning rate was determined as 0.005.

### 4.2.2 Batch-size

When training the CNN, we can not let all the samples be used for network training concurrently, due to the large size of sample data, restrictions on computer configuration, and other conditions. Therefore, it is

**Table 3** Loss, accuracy and time cost under different batch-size

Batch-size	Train-loss	Test-loss	Train-acc	Test-acc	Time cost(s)
1	0.0597	0.000074	0.9788	1	828
2	0.1105	0.000076	0.9552	1	484
4	0.1369	0.000076	0.9477	1	278
5	0.1585	0.000085	0.9391	1	240
10	0.2468	0.0031	0.9123	1	154
20	0.7756	0.0306	0.7314	0.999	123
40	0.7718	0.0456	0.7632	0.9774	77
50	1.1889	0.3048	0.59	0.9732	73
80	1.3441	0.8301	0.5319	0.7982	62
100	1.5499	1.4332	0.3985	0.5704	60
200	1.5572	1.48	0.421	0.556	52
400	1.5767	1.5119	0.3995	0.6136	48
500	1.593	1.5536	0.3446	0.548	50

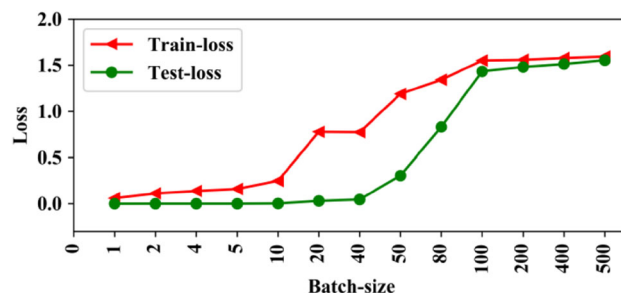
usually preferable to divide the sample into blocks of moderate size. The size of this block is called batch-size [27]. In this experiment, we used different batch-sizes to train the CNN. The other parameters were the same, and the iteration was 1. The loss, accuracy, and time cost are shown in Table 3.

From Figures 8, 9, and 10, it can be seen that when the batch-size is very small, the network training and testing accuracy is very high. However, it takes a longer time to carry out one iteration. As the batch-size becomes larger (especially when the batch-size is greater than 20), training and testing accuracy is progressively reduced; however, the time required for one iteration is also progressively reduced. By carrying out a comprehensive comparison with regard to the above situation, we found that when the batch-size is 20, we could not only ensure accuracy, but also reduce training time.

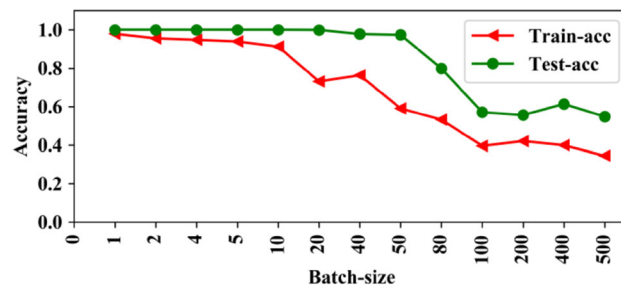
Finally, the pre-processed time-frequency map is input into the CNN model (learning rate is 0.005, and batch-size is 20) as shown in Figure 3. Figure 11 shows the training loss and accuracy after many iterations. Diagnostic accuracy becomes 100%. This shows that the STFT method combined with CNN, can effectively identify motor faults and achieve diagnose them intelligently.

### 4.3 Investigation of Feature Visualization

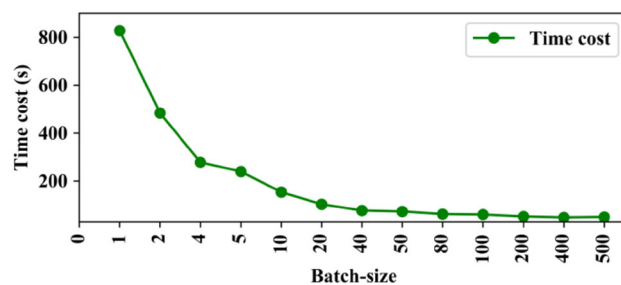
To further clarify the advantages of the CNN, we carried out feature visualization with the input samples of different faults using the trained network. Here, we selected done sample for each fault, as shown in Figure 12. Then, we put these seven samples into the trained CNN to make a prediction.



**Figure 8** Loss under different batch-size



**Figure 9** Accuracy under different batch-size



**Figure 10** Time cost under different batch-size

#### 4.3.1 Investigation of Convolutional Layer

The first convolutional layer filters the  $100 \times 100 \times 3$  input image with sixteen  $3 \times 3 \times 3$  sized kernels, with a stride of 1 pixel (this is the distance between the receptive field centers of neighboring neurons in a kernel map). We printed the 16 kernels in the first convolutional layer (Figure 13). Due to the smaller size of the convolution kernel, it is difficult to find practical significance; therefore, we printed the feature maps exported from the first convolutional layer. As shown in Figure 14, each feature map corresponds to a different fault; therefore, it was easy to find the difference between these maps. Consequently, the 16 convolution kernels can effectively extract different features of different faults. In addition, we also printed the feature maps obtained by the second convolutional layer (Figure 15). From these 7 maps, we can clearly find their differences. It is obvious that the convolution kernels (filters), which were obtained by network training, are

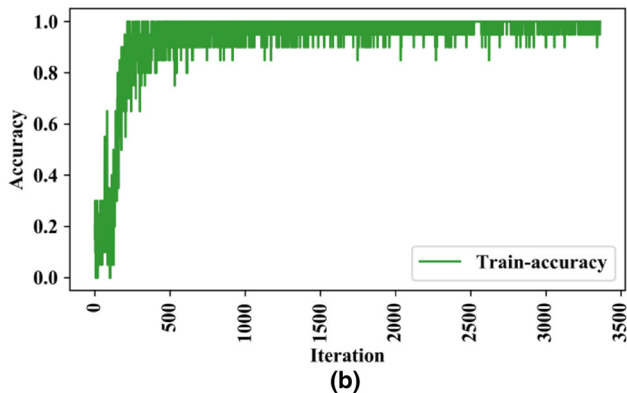
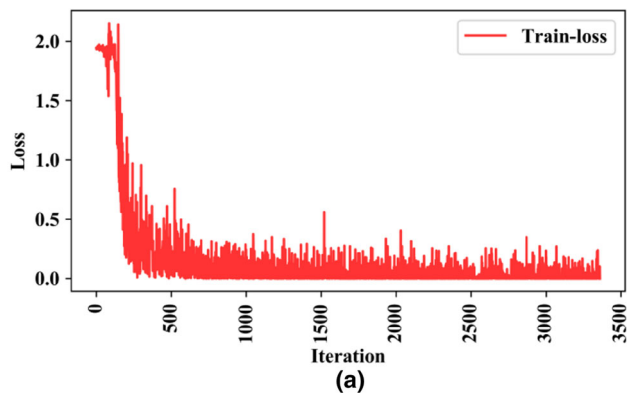


Figure 11 Training loss and accuracy

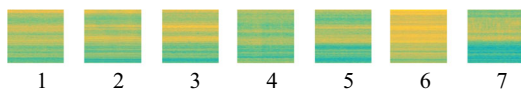


Figure 12 The 7 fault samples

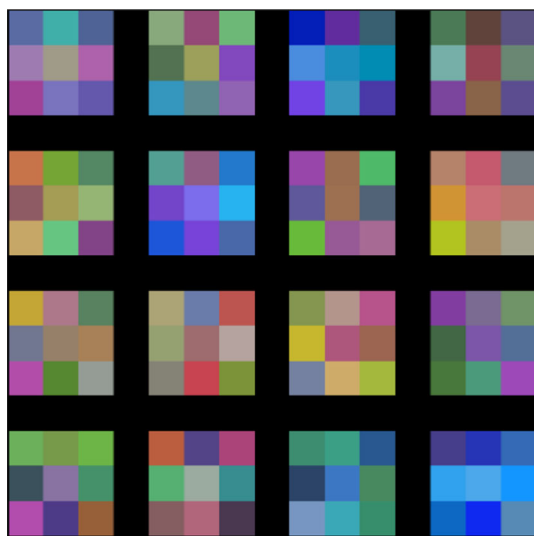


Figure 13 16 convolutional  $3 \times 3 \times 3$  sized kernels learned by the first convolutional layer on the  $100 \times 100 \times 3$  input images

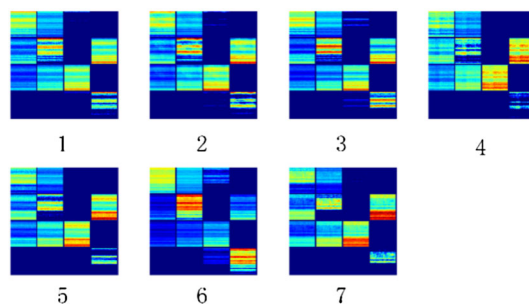


Figure 14 Feature maps of  $C_1$

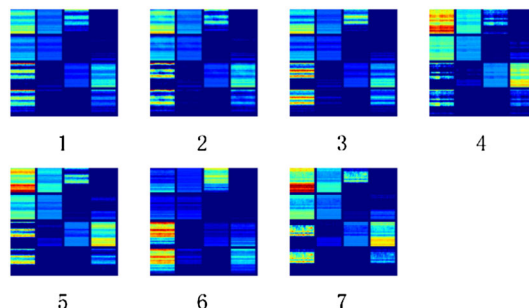


Figure 15 Feature maps of  $C_2$

effective, and that they learned the features in the input images. They were able to map different feature maps for different faults.

### 4.3.2 Investigation of Fully Connected Layer

To study the features extracted by the fully connected layer, we took out the output of the first fully connected layer, which had 256 neurons, under the seven samples, and obtained seven  $1 \times 256$  vector sets, corresponding to seven types of faults. Then, we sought the Pearson correlation coefficient between any two of these 7 vectors. The result is shown in Table 5, and the Pearson correlation coefficient [28] is expressed by Eq. (6).

$$r = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{\sqrt{\sum (X - \bar{X})^2 \sum (Y - \bar{Y})^2}}, \tag{6}$$

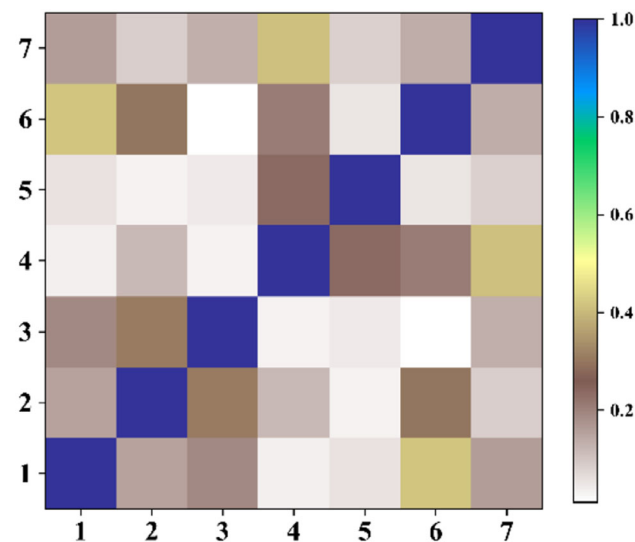
where  $X$  and  $Y$  represent the two vectors used for comparison. To facilitate observation, we obtain the absolute value of the correlation coefficient.

From Table 4, we excluded the correlation coefficient between the same diagonal samples. The maximum correlation coefficient is only 0.42 and represents moderate correlation. Next, we drew a heat-map (Figure 16) using the data in Table 4. By combining the color distribution in the heat-map, it can be seen that the Pearson correlation coefficients are generally lower than 0.3, and that a



**Table 4** Pearson correlation coefficient with output of 7 fully connected layer samples

Sample	1	2	3	4	5	6	7
1	1.00	0.15	0.19	0.04	0.06	0.42	0.16
2	0.15	1.00	0.31	0.12	0.03	0.30	0.09
3	0.19	0.31	1.00	0.03	0.05	0.01	0.13
4	0.04	0.12	0.03	1.00	0.24	0.21	0.41
5	0.06	0.03	0.05	0.24	1.00	0.05	0.08
6	0.42	0.30	0.01	0.21	0.05	1.00	0.14
7	0.16	0.09	0.13	0.41	0.08	0.14	1.00



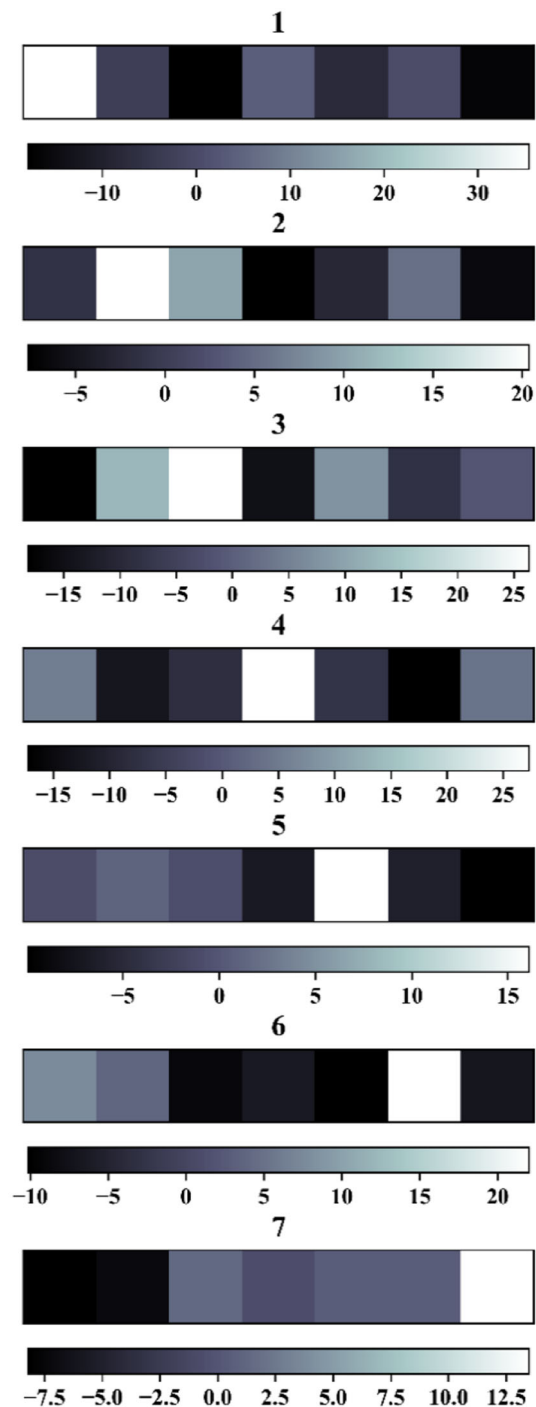
**Figure 16** Pearson correlation coefficient with output of 7 fully connected layer samples (0.0–0.2: extremely weak correlation or no correlation; 0.2–0.4: weak correlation; 0.4–0.6: moderate correlation; 0.6–0.8: strong correlation; 0.8–1: extremely strong correlation)

significant part of them is lower than 0.1. Obviously, there is a considerable gap between the output features, if different faults exist in this fully connected layer.

Finally, because the training set label used one-hot coding, and the sample classification number was 7, in the  $1 \times 7$  vector of the fully connected layer output, the position of the maximum component is the result of network output. Therefore, we took out the output feature of the last fully connected layer with 7 samples. Then, we drew the heat-map of each sample (Figure 17). It is obvious that the position of the maximum component corresponds the correct sample classification result.

#### 4.4 Comparative Analysis

Feature extraction and pattern recognition are the two main processes of motor fault diagnosis. Presently, the main methods of feature extraction are wavelet analysis, empirical mode decomposition, and principal component



**Figure 17** The 7 output features of the last fully connected layer with 7 samples

analysis. We can also carry out feature extraction by analyzing the average motor signal variance, kurtosis, peak value, and energy ratio. The methods commonly used in motor fault diagnosis are BP neural network and SVM.

In this study, for comparison to traditional intelligent methods, we used empirical mode decomposition (EMD) + SVM [29], PCA + SVM [30], and diagnosis features +

**Table 5** Results of different fault diagnosis methods for motors

Method	Sample	Sample number	Diagnostic accuracy
PCA + SVM	Time domain signal	9100	30.52%
EMD + SVM	Time domain signal	9100	93.67%
Diagnostic Features + SVM	Time domain signal	9100	95.05%
SDAE	Frequency signal	9100	99.9048%
STFT + CNN	Time spectrum diagram	14000	100%
Wavelet + CNN	Time spectrum diagram	14000	100%

SVM, to carry out motor fault diagnosis. Moreover, the stacked de-noising auto-encoder [21] (SDAE) may also be used, for contrast, and as a common time-frequency analysis method, the wavelet transform + CNN method may also be used for comparison. The results are given in Table 5.

PCA is essentially a linear method, which is weak in dealing with nonlinear problems. Therefore, the PCA and SVM method had an unsatisfactory effect and its diagnostic accuracy was only 30.52%. EMD can adaptively decompose signals. The diagnostic accuracy of EMD + SVM, and diagnostic features + SVM, were 93.67% and 95.05%, respectively. In the above methods, the radial basis function (RBF) was used. In the field of deep learning, SDAE is a popular unsupervised network model. We used different fault frequency domain signals as input samples and the length of each input was 2000. The SDAE network structure was [2000, 100, 100, 50, 7], with three hidden layers, the number of nodes in each layer was 100, 100, and 50, respectively. The used batch-size was 35, the sparsity criterion was 0.1, and the learning rate was 0.5. Due to the deep learning model's strong representation capability, the accuracy of the SDAE in the data set was also very high, reaching up to 99.9048%. We also used the wavelet transform, using the Morlet wavelet with a bandwidth parameter and a center frequency of 4, to transform the original signal into a time spectrum diagram. Then, we used these pre-processed data to train the same CNN network. The test set classification results also reached 100%. Thereby, we proved the effectiveness of the method in converting the one-dimensional signal to a time spectrum diagram.

The running time of the algorithm is an important parameter. For traditional feature extraction methods, the process of feature extraction often consumes a lot of time. Thanks to the low dimensionality of the extracted features, the training time of the traditional classifier was very short. However, in deep learning methods, the input data of the network usually performs only simple pretreatment, and,

therefore, the dimension was very large. Moreover, the network has a lot of weight; therefore, The SDAE method takes 1030 s for 300 iterations. Due to using a GPU for training the CNN, the training time of the CNN was reduced to an acceptable value.

#### 4.5 Industrial Implementation Plan

In this paper, a method for offline fault diagnosis was described. The object of this experiment was the drivetrain diagnostics simulator system. When the diagnostic object was changed to other motors, the signal of the motor under different faults was required. Then, these data were used as the input to fine-tune the trained CNN. If we can obtain different fault signals from different types of motors, and put this huge amount of data into a deeper and more complex network, then the network model will be more meaningful. Due to the huge amount of data, the training of the CNN may be time-consuming; however, it is worthwhile to invest a significant amount of time in the training of an excellent network model. Moreover, when using the trained model to classify the unknown signal, only one forward operation is required. Under our experimental environment, it takes only 2 s to complete the classification of the 2800 test samples. Therefore, the trained network can be used in industrial applications.

## 5 Conclusions

In this paper, we reported a deep learning method for motor fault diagnosis using a CNN. Prior to network training, we preprocessed the original signal and converted it to a time-frequency map by STFT. By selecting different training parameters, an optimal one could be obtained to achieve a test set accuracy of up to 100%. Through the diagnosis results of comparative experiments, we found that the proposed deep learning method was able to adaptively mine salient fault characteristics and effectively identify

the health states with high diagnosis accuracy. After investigating feature visualization and carrying out a comparison against traditional diagnosis algorithms, the main advantage of the proposed method was found to be that the fault features were learned via a general-purpose learning procedure, instead of being hand-engineered or having prior knowledge of signal processing techniques, which is easy to apply to diagnosis issues.

Future work will include more experimental tests to further understand the limitations of the SFFT and CNN methods, particularly regarding some more complex faults, such as rolling bearings, gears, and even composite faults. Additionally, in this study, we used a fixed network architecture. However, this is still an open problem for optimal parameter determination, particularly when a deeper architecture is employed, or another completely different fault is investigated.

## References

- Chuan Li, R V Sanchez, G Zurita, et al. Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals. *Mechanical Systems and Signal Processing*, 2016, 76–77: 283–293.
- Ya-Guo Lei, Nai-Peng Li, Jing Lin, et al. Two new features for condition monitoring and fault diagnosis of planetary gearboxes. *Journal of Vibration & Control*, 2013, 21(4): 755–764.
- Jun-Jian Hou, Wei-Kang Jiang, Wen-Bo Lu. Application of a near-field acoustic holography-based diagnosis technique in gearbox fault diagnosis. *Journal of Vibration & Control*, 2013, 19(1): 3–13.
- A M D Younus, B S Yang. Intelligent fault diagnosis of rotating machinery using infrared thermal image. *Expert Systems with Applications*, 2012, 39(2): 2082–2091.
- J. R Ottewill, M Orkisz. Condition monitoring of gearboxes using synchronously averaged electric motor signals. *Mechanical Systems & Signal Processing*, 2013, 38(2): 482–498.
- A Contin, S D'orlando, G Fenu, et al. Experiments on actuator fault diagnosis: the case of a nonlinearly controlled AC motor. *Control Conference*, 2015: 2747–2752.
- A Glowacz. DC motor fault analysis with the use of acoustic signals, Coiflet wavelet transform, and K-nearest neighbor classifier. *Archives of Acoustics*, 2015, 40(3): 321–327.
- Hui-Min Zhao, Cai-Hua Fang, Deng Wu. Research on motor fault diagnosis model for support vector machine based on Intelligent optimization methods. *Journal of Dalian Jiaotong University*, 2016, 37(1): 92–96. (in Chinese).
- Ping Li, Xue-Jun Li, Ling-Li Jiang, et al. Fault diagnosis of asynchronous motor based on KPCA and PSOSVM. *Journal of Vibration Measurement & Diagnosis*, 2014, 34(4): 616–620. (in Chinese).
- D H Pandya, S H Upadhyay, S P Harsha. Fault diagnosis of rolling element bearing by using multinomial logistic regression and wavelet packet transform. *Soft Computing*, 2014, 18(2): 255–266.
- M Khazaei, H Ahmadi, M Omid, et al. Classifier fusion of vibration and acoustic signals for fault diagnosis and classification of planetary gears based on dempster-shafer evidence theory. *ARCHIVE Proceedings of the Institution of Mechanical Engineers Part E Journal of Process Mechanical Engineering*, 2014, 228(1): 21–32.
- G E Hinton, S Osindero, Y W Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 2006, 18(7): 1527–1554.
- G Hinton, Deng Li, Yu Dong, et al. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Processing Magazine*, 2012, 29(6): 82–97.
- A Krizhevsky, I Sutskever, G E Hinton. Imagenet classification with deep convolutional neural networks. *International Conference on Neural Information Processing Systems*, 2012, 25(2): 1097–1105.
- Yan-Feng Li, Xin-Qing Wang, Mei-Jun Zhang, et al. An approach to fault diagnosis of rolling bearing using SVD and multiple DBN classifiers. *Journal of Shanghai Jiaotong University*, 2015, 49(5): 681–686. (in Chinese).
- Jia Feng, Ya-Guo Lei, Jing Lin, et al. Deep neural networks: a promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems & Signal Processing*, 2016, 72–73: 303–315.
- I Arel, D C Rose, T P Karnowski. Research frontier: deep machine learning—a new frontier in artificial intelligence research. *IEEE Computational Intelligence Magazine*, 2010, 5(4): 13–18.
- P Tamilselvan, Ping-Peng Wang. Failure diagnosis using deep belief learning based health state classification. *Reliability Engineering & Systems Safety*, 2013, 115(7): 124–135.
- P Vincent, H Larochelle, I Lajoie, et al. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 2010, 11(12): 3371–3408.
- Zhao-Feng Zhang, Long-Biao Wang, A Kai, et al. Deep neural network-based bottleneck feature and denoising autoencoder-based dereverberation for distant-talking speaker identification. *Eurasip Journal on Audio, Speech, and Music Processing*, 2015, 2015(1): 1–13.
- Chen Lu, Zhen-Ya Wang, Wei-Li Qin, et al. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification. *Signal Processing*, 2017, 130: 377–388.
- S H Nawab, T F Quatieri. Short-time fourier transform. *Advanced Topics in Signal Processing*, 1988, 32(2): 289–337.
- S Lawrence, C L Giles, A C Tsoi, et al. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 1997, 8(1): 98–113.
- Wei-Yang Liu, Yan-Dong Wen, Zhi-Ding Wen, et al. Large-margin softmax loss for convolutional neural networks. *International Conference on International Conference on Machine Learning*, 2016: 507–516.
- D Erhan, Y Bengio, A Courville, et al. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 2010, 11(3): 625–660.
- I Daubechies. The wavelet transform, time-frequency localisation and signal analysis. *IEEE Transactions on Information Theory*, 1990, 36(5): 961–1005.
- S Ioffe, C Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. *Computer Science*, 2015: 448–456.
- P Ahlgren, B Jarneving, P B R Ahlgren. Requirements for a cocitation similarity measure, with special reference to Pearson's correlation coefficient. *Journal of the American Society for Information Science & Technology*, 2003, 54(6): 550–560.
- Shu-Fang Li, Wei-Dong Zhou, Qi Yuan, et al. Feature extraction and recognition of ictal EEG using EMD and SVM. *Computers in Biology & Medicine*, 2013, 43(7): 807–816.
- Neng Ren. PCA-SVM-based automated fault detection and diagnosis (AFDD) for vapor-compression refrigeration systems. *Hvac & R Research*, 2010, 16(3): 295–313.

**Li-Hua Wang**, born in 1974, is currently a Senior laboratory technician at *School of Information and Control, Nanjing University of Information Science and Technology, China*. Her research interests include fault diagnosis, pattern recognition and signal processing. E-mail: wlh\_nj@163.com.

**Xiao-Ping Zhao**, is currently an associate professor at *Nanjing University of Information Science and Technology, China*. She received her doctor degree from *Nanjing University of Aeronautics and Astronautics, China*, in 2009. Her research interests include signal processing and fault diagnosis. E-mail: zxp@nuist.edu.cn.

**Jia-Xin Wu**, born in 1993, is currently a master candidate at *School of Computer and Software, Nanjing University of Information Science*

*and Technology, China*. His research interests include deep learning, fault diagnosis. E-mail: 799048410@qq.com.

**Yang-Yang Xie**, born in 1991, is currently a master candidate at *Nanjing University of Information Science and Technology, China*. His research interests include fault diagnosis pattern recognition. E-mail: 923819548@qq.com.

**Yong-Hong Zhang**, born in 1974, is currently a professor and a PhD candidate supervisor at *Nanjing University of Information Science and Technology, China*. He received his PhD degree from *Shanghai Jiao Tong University, China*, in 2005. His research interests include precision instruments and machinery, pattern recognition and intelligent systems, remote sensing information processing. E-mail: zyh@nuist.edu.cn